SPORTS BUDDY

A Web-Based Sports Event Management System

Submitted By: Mohit

GitHub Repository:

https://github.com/mohit00m/sports-buddy-unified-mentors.git

1. Introduction

Sports Buddy is a web-based sports event management system developed using HTML, CSS, JavaScript, and Firebase.
The system allows users to register, login, and explore sports events, while administrators can manage categories,
cities, areas, and events through a secure dashboard interface.

2. Objectives

- Develop secure authentication using Firebase Authentication
- Implement role-based access control (Admin/User)
- Provide CRUD operations for sports events
- Maintain structured Firestore database collections
- Deploy the system using Firebase Hosting

3. Technology Stack

Frontend Technologies:
- HTML5
- CSS3
- JavaScript (ES6 Modules)

Backend Services:
- Firebase Authentication
- Firebase Firestore Database
- Firebase Hosting

4. System Architecture

The system follows a client-cloud architecture:

User Browser → Firebase Authentication → Firestore Database → Firebase Hosting

The frontend interacts securely with Firebase services. Role-based rules ensure data protection and controlled access.

5. Database Design

Firestore Collections Used:

- users
- events
- categories
- cities
- areas
- logs

Each collection is designed for scalability and proper normalization.

6. Modules Description

User Module:

- Register
- Login
- View Events
- Logout

Admin Module:

- Add Events
- Update Events
- Delete Events
- Manage Categories
- Manage Cities
- Manage Areas
- View Dashboard Statistics

Logging Module:

- Tracks event creation, update, deletion
- Maintains activity history

7. Security Implementation

- Firebase Authentication ensures only verified users can access the system.
- Role-based Firestore rules restrict write operations to admin users only.
- Unauthorized access redirects to login page.
- Logging system records administrative actions.

8. Code Optimization

Code Level Optimization:
- Modular ES6 structure
- Reusable functions
- Async/Await for asynchronous operations
- Centralized Firebase configuration

Architecture Level Optimization:
- Clean separation of concerns
- Role-based access enforcement
- Structured Firestore collections
- Cloud-based scalable deployment

9. Testing and Test Cases

Functional Test Cases:

- Valid user registration → Account created successfully
- Login with correct credentials → Dashboard loads
- Login with incorrect credentials → Error message displayed
- Add event as admin → Event stored in Firestore
- Update event → Changes reflected
- Delete event → Event removed
- Unauthorized user attempting edit → Access denied

Security Testing:
- Firestore rules block unauthorized writes
- Non-authenticated users redirected to login

10. Deployment

The project is deployed using Firebase Hosting.

Deployment Steps:
1. Firebase CLI installed
2. Project initialized using firebase init
3. Hosting configured
4. Project deployed using firebase deploy

Live Application:
https://sportsbuddy-mohit-2026.web.app

11. Challenges Faced

- Firestore security rule configuration
- ES6 module import issues
- Handling asynchronous data operations
- Designing professional UI layout

12. Future Enhancements

- Event filtering by location
- Search functionality
- Admin analytics dashboard
- Real-time notifications
- Sports matching algorithm

13. Conclusion

Sports Buddy successfully implements a secure, scalable, and professional sports event management platform.
The system demonstrates authentication, role-based security, optimized architecture, and cloud deployment,
meeting all functional and non-functional project requirements.